UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/724,588	11/28/2003	Laurence B. Boucher	ALA-025	9422

24501 7590 11/01/2006

EXAMINER	

MARK A LAUER
6601 KOLL CENTER PARKWAY
SUITE 245
PLEASANTON, CA 94566

ART UNIT	PAPER NUMBER

DATE MAILED: 11/01/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 10/724,588
Filing Date: November 28, 2003
Appellant(s): BOUCHER ET AL.

MAILED

NOV 01 2006

Technology Center 2100

Mark Lauer

For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 8/2/2006 appealing from the Office action mailed 1/18/2006.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

Bennett et al.	U.S. Pat No. 6345302	02-2002

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-8, 10-14, 16-27, 29-40 and 42 are rejected under 35 U.S.C. 102(e) as being anticipated by Bennett et al. [U.S. Pat. No. 6345302].

As to claim 1, Bennett teaches the invention as claimed including: a system for communication by a local host that is connectable by a network to a remote host [e.g., 276, 1000, Fig.10], the system comprising:

a communication processing device (CPD) [2000, Fig.3, which is a network interface card] that is integrated into the local host to connect the network and the local host, said CPD including hardware configured to analyze Internet Protocol (IP) and Transmission Control Protocol (TCP) headers of network packets [Abstract; col.15, lines 52-61; i.e., verify the TCP checksum, which is entered in a field of the header (see 322, Fig.7)]; and

a central processing unit (CPU) [10, Fig.3] running protocol processing instructions in the local host to create a TCP connection between the local host and the remote host [col.4, lines 23-50; note that the "TCP connection" is being construed as a physical connection established for transmission of a TCP packet. Since TCP process 91 of Fig. 2B is designed for constructing a TCP

packet, the execution of process 91 is in a way "running protocol processing instructions"], said CPU providing to said CPD a media-access control (MAC) address, an IP address and a TCP port that correspond to said connection [e.g., col.4, lines 7-22; col.16, lines 19-35; Figs. 2A, 6-7; wherein Figs. 6-7 show that the encapsulated headers of an outbound TCP/IP packet contain TCP port and IP addresses, whereas an outbound ARP (Address Resolution Protocol) request must contain at least a source MAC address and a target IP address (see also col.11, line 33 – col.12 line 20, wherein the MAC address is used in an Ethernet environment and the ARP is an inherent function of an Ethernet node). Both the TCP/IP packet and ARP request frame are prepared at the host CPU and handed down to the CPD for transmission],

wherein said CPD and said CPU are configured such that a message transferred between the network and the local host is generally processed by said CPD instead of said CPU when said CPD controls said connection and said message corresponds to said connection [col.4, lines 60-65; col.12, lines 21-37; col.16, lines 4-35; col.25, lines 32-34; i.e., the message corresponds to the outbound ACK packets and the inbound ACK packets; both are processed within the CPD without involvement of the host CPU, wherein transmission of the inbound and outbound ACK occurs when the CPD has control of the TCP connection (because the CPU is not involved in these activities)].

As to claims 2-3, Bennett further teaches that said CPU provides to said CPD an address in local host memory for storing or retrieving application data from said message [col.8, lines 24-32; note that the DMA controller must obtain an address from the CPU (via device driver) in order to perform necessary data movement].

As to claim 4, Bennett further teaches that said CPD is connected to said CPU by a bus [37, Fig.3].

As to claim 5, Bennett further teaches that said CPD includes a microprocessor [col.9, lines 37-43].

As to claims 6-7, Bennett further teaches that said CPD is connected to an input/output (I/O) controller [col.8, lines 2-11].

As to claim 8, Bennett teaches that the system further comprises a memory that is disposed in said host and accessible by said CPU and said CPD [e.g., 114, Fig.3].

As to claim 10, Bennett further teaches that said CPD is integrated with a peripheral component interconnect (PCI) bridge [e.g., two PCI bridges 9080 are integrated with the network processor in Fig.9].

As to claim 11, Bennett further teaches that said CPD is integrated with a memory controller for said CPU [e.g., 926, 928, Fig.9; col.9, lines 44-52].

As to claim 12, Bennett further teaches that said CPD is integrated with an I/O controller [e.g., 912, 914, Fig.9] and a memory controller [926, 928, Fig.9] for said CPU.

As to claim 13, The system of claim 1, wherein said CPD is connected with an I/O controller that connects said CPD to a memory controller for said CPU [Fig.9; col.24, lines 13-64].

As to claim 14, Bennett further teaches that said CPD is connected to a hub interface bus that connects a memory controller to an I/O controller [e.g., 30, Fig.3; col.5, lines 11-18].

As to claim 16, Bennett further teaches that said message is received from the network by the local host [col.21, lines 3-37; note that the message in this passage is transferred from the remote node].

As to claims 17-27 and 29-40, since the features of these claims can also be found in claims 1, 4-6, 8, 10-14 and 16, they are rejected for the same reasons set forth in the rejection of claims 1, 4-6, 8, 10-14 and 16 above.

Note that the "second network packet" of claim 30 is mapped to Bennett's ACK packet (see the passages cited in the rejection of claim 1 above) and the term "classifying a second network packet" is being construed as "identifying an ACK packet" in accordance with Bennett's teachings.

As to claim 42, Bennett further teaches that said second network packet is received from the network by the local host [e.g., col.25 lines 32-34].

Claims 28 and 41 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

(10) Response to Argument

With respect to the prior art used by the examiner for rejection, Applicant argues that Bennett is non-enabled because Bennett claims to automatically send an acknowledgement (an "ACK") for a datagram upon verifying the checksum for the datagram, and by doing so Bennett would not be able to carry out TCP's sliding window algorithm, which requires to only acknowledge packets that are contiguous to those previously received.

The argument is not persuasive because: (1) Bennett discloses an ACK format including a "WINDOW" parameter [see Table 1; col. 16 line 36 – col.7 line 7; and 320, Fig.7] in accordance with the typical TCP protocol as taught in RFC 793 [e.g., col.4, lines 16-22]; (2) Bennett is also aware of the issue of "out of order" packets and provides a buffer to temporarily store these types of packets for subsequent reassembling [e.g., col.11 line 66 – col.12 line 6; claims 42 and 53, which shows that reception of an out of order packet is not acknowledged until all the relevant packets are reassembled and placed in their sequential order]; and (3) there is nowhere in the entire disclosure (including the specification, claims and drawings) Bennett explicitly teaches to issue an ACK to any noncontiguous TCP packet [in particular, the scenarios illustrated in Fig. 12 and claim 43 (at col.25 lines 32-34) are consistent with this observation. And, as a matter of fact, even if Bennett chooses to use the much criticized "stop-and-wait"

algorithm, it is still an appropriate TCP protocol because it satisfies the sliding window algorithm by fixing the window size to one; it may be inefficient, but definitely not non-enabled]. The point is: to assert that a prior art is non-enabled one must find explicit evidence to support such allegation without intentional distortions. Applicant is found to have overly emphasized the phrase "automatically writes an ACK" [see col.12, lines 10-11] without considering the fact that the ACK commands are "automatically" written into a command list buffer for further consideration. For example, some ACK commands could be subsequently suppressed [see, e.g., claims 8, 11, 17, 24, 26, 29 and 33].

Applicant uses the passages at col.12 lines 7-11 and col.16 lines 19-26 as basis for the argument. Applicant is reminded that at col.12 lines 7-11 Bennett teaches that the condition to automatically send an ACK command to a command list [notice that the command list is stored in a buffer for later consideration] is "reception of a complete IP datagram and a valid checksum result" (underline emphasis added), while at col.16 lines 19-26, Bennett uses "[u]pon successful defragmentation of the datagram and validation of all applicable checksums" (underline emphasis added) as preconditions for sending an ACK. Since out of order packets are resulted in a buffer as depicted at col.11 line 66 – col.12 line 6, it is clear that these successfully received and strictly validated for all applicable checksums packets, for which Bennett's system generates their respective ACK commands by way of either automatically forming at the network card [col.12 lines 7-11] or composing at the host CPU [col.12 lines 11-20], are in fact contiguous, and consequently the acknowledgement itself is cumulative.

As another point of hint, Applicant is further reminded that since Bennett's system is focused on using a network interface card to replace the role of the host CPU in the ACK packet

generation, it is a common sense that whatever algorithm works at the host CPU for the job should be implemented in the network card as well. Thus there is no need to disclose all the implementation details that a skilled artisan has already known. Bennett's teaching may have been incomplete in a sense that it does not explicitly discuss the issue of lost packets and thereby may cause certain readers to be "misguided", but it is definitely applicable at least to the exemplified scenarios wherein issuing ACK packets and sending subsequent packets are nicely interleaved [see Fig. 12 and claim 43].

For at least the above reasons, it is submitted that Bennett is enabled.

As to claim 1, Applicant argues that (1) Bennett's CPD (i.e., the network card) does not control the TCP connection; (2) Bennett's CPU does not provide the CPD a media access control (MAC) address; and (3) Bennett's CPU does not create a TCP connection between the local host and the remote host.

The arguments are not deemed to be persuasive.

As to point (1), it is noted that Applicant does not have a clear and precise definition as to what is meant to be a "TCP connection", the term has thus been intuitively construed as a physical connection established for transmission of a TCP packet. That is, between the host CPU and the network card, whoever is granted (or arbitrated) to independently transmit/receive a TCP packet via the established connection must have controlled the TCP connection at least for that particular moment. Since Bennett's network card has the capability of independently sending/receiving an ACK packet (which by itself is a TCP packet), the network card must have, at least for the particular moment, controlled the TCP connection.

As to point (2), it is noted that Bennett's teachings apply to any computer nodes illustrated in Figs. 1 and 10 including, for example, Ethernet nodes 272 and 276 [col.3 lines 45-59] wherein an Ethernet interface [51, Fig.11A] is built into each network card. Frames or packets resulted from higher level activities such as TCP, ICMP and ARP protocol processing [e.g., 90, Fig.2A, wherein ARP is listed in the transport layer of Ethernet node 276] are prepared at the host CPU except for the portions that Bennett explicitly teaches to be processed at a special hardware in the network card. This ARP data flow assures that at least a source MAC address is included in the Ethernet-bound ARP request frame. Although Bennett does not specifically teach the operation of an ARP, it is submitted that ARP is an inherent function to Bennett's Ethernet nodes such as nodes 272 and 276 and that the Ethernet nodes in Bennett's Figs. 1 or 10 must have provided an MAC address to the CPD in an ARP request frame.

As to point (3), it is noted that Applicant does not specifically teach what is meant by "create a TCP connection" [note that creating a "connection context" is different from creating a connection], the term is being construed as "schedule for transmitting a TCP packet over a physical connection" or "cause a TCP transmission to happen". And for reasons shown in point (1) above, it is submitted that Bennett's CPU and network card are intermittently scheduled to performing TCP transmissions (or occupying a TCP connection), therefore a TCP connection is created for each of the TCP transmission moment. As for the feature of "running protocol processing instructions", it is noted that the passage recited from col.4 lines 22-50 clearly shows an activity engaging a set of protocol processing instructions necessary for preparing a TCP packet (e.g., via processes 91 and 96), and releasing or regaining a TCP connection in a time shared manner [i.e., the time sharing scheme for sharing the TCP connection resource between the CPU and the CPD is a protocol, and the act of such a time sharing scheme is an act of "running protocol processing instructions"].

As to claims 17, 30 and 42, Applicant argues that the features of these claims are not all covered by claims 1, 4-6, 8, 10-14 and 16 and therefore rejection of claims 17, 30 and 42 by simply referring to the rejection of claims 1, 4-6, 8, 10-14 and 16 is improper. For example, Applicant argues that: (1) claim 30 has an additional limitation: "said CPD receiving control of said connection from said CPU, said CPD classifying a second network packet as corresponding to said connection and processing said second network packet without any protocol processing of said second network packet by said CPU" that is not found in claims 1, 4-6, 8, 10-14 and 16 because Bennett does not teach that processing of the second network packet (to which the examiner equates the ACK packet) is independent of the CPU. Applicant cites col. 16 lines 19-35 to show that Bennett's the protocol logic state must be maintained and controlled elsewhere. Applicant further argues that "the second network packet" is not found in Bennett; (2) as to claim 17, Applicant repeats the argument about the features of providing an MAC address from the CPU to the CPD and running protocol processing instructions by a CPU to create a TCP connection being missing from Bennett. Applicant further attacks that since Bennett is not enabled, a TCP connection could not have been reliably created; and (3) as to claim 42, Applicant argues that Bennett does not teach receiving a second network packet that is independently processed at the CPD without any protocol processing of the packet by the CPD.

Applicant's argument is not deemed to be persuasive.

As to point (1): Applicant is reminded that at col. 16 lines 19-35, which Applicant's argument is based upon, Bennett further teaches that "[t]he saved data includes source IP address, datagram sequence identification number, source TCP port number, destination port number, and the available datagram memory (used for window size). These values are stored by TCP logic 93 in command list 42, which is resident in protocol logic 45." Among the listed items, source IP address, datagram sequence identification number, source TCP port number, destination port number are directly obtainable from the

datagram headers [see Figs. 6-7]. The only items that are needed for constructing an ACK packet and are not obtainable from the datagram headers are the acknowledgement sequence number and the window size [see Fig.7 for TCP header format and related activities at col. 20 lines 28-64], wherein the acknowledgement sequence number is obtainable from the received TCP packet to which the ACK is responding (i.e., one plus the sequence number field of the segment being acknowledged) and the window unallocated size is the space in buffer 53 of Fig. 4 (the network card) [col.16 lines 67 – col. 17 line 7]. Both items are being referred to as "protocol logic state", and it clearly shows that both items are held within the network card [note that all TCP segments and IP datagram are received at the network card for validating the checksums].

With respect to the feature of "said CPD classifying a second network packet": it has been pointed out in the previous office action that the second network packet corresponds to Bennett's "ACK packet" which is being singled out (or "classified") as one kind of packet that can be independently processed by the CPD. To simply put it, the examiner maps "said CPD receives control of said connection, said CPD classifying a second network packet as corresponding to said connection and processing said second network packet without any protocol processing of said second network packet by said CPU" of claim 30 to "a message … generally processed by said CPD instead of said CPU when said CPD controls said connection and said message corresponds to said connection" of claim 1. Note that due to the lack of specific definition in Applicant's specification, the word "classifying" does not carry any special meaning other than "identifying" or "specifying". Thus, a message picked up by the CPD and processed by said CPD instead of said CPU (claim1) is mapped to the second network packet that is identified by the CPD and processed independently therein (claim 30). These features in claims 1 and 30 may have been worded differently, but they carry the same meaning. And in particular when these features are mapped to Bennett, the ACK packet stands out as being identified by the CPD to be

independently processed in it, wherein the CPD gains control of the TCP connection (from the CPU) when the ACK packet is to be transmitted from the CPD to the remote node.

With respect to the term "second network packet", Applicant might have something else in mind such as "a packet coming from a second network" (i.e., other than a first network from which the first network packet originates). However, such interpretation could not be supported because the claim language requires that both the first and second network packets be originated from the same remote node and correspond to the same connection.

As to point (2): see the examiner's response to Applicant's argument points (2)-(3) related to claim 1 above.

As to point (3): see the rejection of claim 42 in this office action, wherein Applicant is directed to a passage at col.25 lines 32-34 that clearly show the reception of an inbound ACK packet.

As to claims 28 and 41, Applicant's arguments are persuasive, therefore the rejection of these claims are withdrawn.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Wen-Tai Lin 10/30/06

Wen-Tai Lin

Conferee:

John Follansbee

JOHN FOLLA[illegible]
SUPE[illegible] EXAMINER
TE[illegible] CENTER 2100

DAVID WILEY
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100